SPREAD SPECTRUM CLOCKING FOR DATA TRANSFER BUS LOADING

BACKGROUND

[0001] Spread spectrum clocking spreads the energy of a signal over a range of frequencies to dissipate the energy rather than having the signal concentrated at a particular constant frequency. Spread spectrum clocking can be accomplished by modulating the phase (e.g., dithering) the control clock timing of the signal. Spreading the energy of a signal over a range of frequencies can help to reduce electromagnetic interference emissions and noise associated with a signal that would otherwise be concentrated at one frequency.

[0002] Data transfer buses in computing devices, such as a data bus, address bus, control bus, and the like, communicate or transfer data between components in the computing device. For example, a microprocessor can be coupled to various memory devices via a memory bus. The number of devices that can be coupled to communicate via a data transfer bus at any one time is determined by bus loading which describes the limit of signal load capacity for a particular bus. Data transmissions that exceed the capacity of a data transfer bus can result in data transmission errors due to timing margin inaccuracies.

BRIEF DESCRIPTION OF THE DRAWINGS

[0003] The same numbers are used throughout the drawings to reference like features and components:

Fig. 1 illustrates an exemplary data transfer circuit in which spread spectrum clocking for data transfer bus loading can be implemented.

Fig. 2 illustrates an exemplary control system in which spread spectrum clocking for data transfer bus loading can be implemented.

Fig. 3 illustrates a data transfer frequency chart that illustrates exemplary spread spectrum clocking for data transfer bus loading.

Fig. 4 is a flow diagram that illustrates an exemplary method for spread spectrum clocking for data transfer bus loading.

Fig. 5 illustrates various components of an exemplary printing device in which spread spectrum clocking for data transfer bus loading can be implemented.

Fig. 6 illustrates various components of an exemplary computing device in which spread spectrum clocking for data transfer bus loading can be implemented.

DETAILED DESCRIPTION

[0004] The following describes systems and methods related to spread spectrum clocking for data transfer bus loading, such as when transferring or communicating data via a data transfer bus. A data transfer bus includes any one of a data bus, an address bus, a control bus, a memory bus, and the like. Spread spectrum clocking is implemented to minimize, or otherwise decrease, electromagnetic emissions associated with transferring data via a data transfer

bus. This decreases the strength of a data signal at any one frequency for regulatory compliance of electromagnetic emissions limits for electronic products and devices. Controlling electromagnetic emissions can also alleviate the need for costly metal housings to shield the emissions of a device from other components, such as computer monitors, telephones, and the like.

[0005] Fig. 1 illustrates an exemplary data transfer circuit 100 in which spread spectrum clocking for data transfer bus loading can be implemented. The data transfer circuit 100 includes a data circuit 102, control logic 104, one or more system components 106, and a data transfer bus 108 over which data circuit 102 communicates data with the one or more system components 106. Data transfer bus 108 can include any one or more of a data bus, an address bus, a control bus, a memory bus, and the like.

[0006] The data circuit 102 includes an operating condition(s) status 110 and a spread spectrum clocking control 112. The operating condition(s) status indicates operating condition(s) of the data circuit 102 and corresponds to data communications loading on the data transfer bus 108. The operating conditions can include any one of process, voltage, and/or temperature conditions of the data circuit 102.

[0007] A process variation can occur when the data circuit 102 is manufactured. For example, data circuit 102 can be an application-specific integrated circuit (ASIC) that has a manufactured silicon variance which affects data communications transfer and bus loading. Further, an increase in voltage applied to data circuit 102 will increase the data transfer speed, but will also increase the operating temperature which decreases the data transfer speed.

Collectively, the process-voltage-temperature (PVT) operating conditions of data circuit 102 affect the bus loading and data transfer via data transfer bus 108.

[0008] The spread spectrum clocking control 112 controls a frequency spread deviation for data communication via data transfer bus 108. The spread spectrum clocking control 112 can be implemented as a phase-locked loop, for example, that dithers the frequency signal of a data communication. The frequency spread deviation of data communications can be controlled by adjusting a minimum clock frequency and a maximum clock frequency, or by adjusting a percentage clock frequency deviation from a center frequency. The minimum clock frequency and the maximum clock frequency define a dithering range to spread out the energy of the communicated data.

[0009] The control logic 104 can be implemented to obtain the operating condition(s) status 110 and generate an input to the spread spectrum clocking control 112 to adjust the frequency spread deviation of data communications according to the operating conditions status 110. The frequency spread deviation (e.g., spread spectrum clocking) can be adjusted to minimize, or otherwise decrease, electromagnetic emissions associated with data communications via data transfer bus 108. Spread spectrum clocking adjustment for data bus loading can be implemented for any type of data communication bus, such as a variable loading data transfer bus that has flexible loading parameters, to minimize electromagnetic emissions when operating conditions of the data circuit 102 are varied.

[0010] Fig. 2 illustrates an exemplary control system 200 in which spread spectrum clocking for data transfer bus loading can be implemented. The control system 200 includes an application-specific integrated circuit (ASIC)

202, firmware 204, a processor (or controller) 206, and a processor bus 208 over which the ASIC 202 transfers data to the processor 206. The control system 200 also includes any number of other data buses, such as memory buses 210(1), ..., 210(N), and memory component(s) 212 and 214. The ASIC 202 can be implemented as a memory controller to transfer data to the memory components 212 and 214 via the memory buses 210(1) and 210(N), respectively.

[0011] The ASIC 202 includes a pressure-voltage-temperature (PVT) status register 216 and a spread spectrum clocking control 218. The PVT status register maintains a PVT status that indicates operating conditions of the ASIC 202 and corresponds to data loading on memory buses 210(1), ..., 210(N). The spread spectrum clocking control 218 controls a frequency spread deviation for data transfer via a memory bus 210. The spread spectrum clocking control 218 controls the frequency spread deviation of data transfer by adjusting a minimum clock frequency and a maximum clock frequency, or by adjusting a percentage clock frequency deviation from a center frequency.

[0012] A memory bus 210 can be expandable and data loading on the bus can depend on any number of memory component 212 variables, such as the size and data transfer speed of a memory component. The ASIC 202 can include multiple strength drive pads which can be adjusted based on the PVT operating conditions of the ASIC 202, and to compensate for an additional load, or increase in data transfer. However, increasing the data transfer loading on a memory bus 210 may also increase the electromagnetic emissions associated with the increased data transfer.

[0013] The firmware 204 includes logic that can be implemented to obtain the PVT operating conditions status from the PVT register 216 and generate an input to the spread spectrum clocking control 218 to adjust the frequency spread deviation of data transfer according to the PVT status. The frequency spread deviation (e.g., spread spectrum clocking) can be adjusted to minimize, or otherwise decrease, the electromagnetic emissions associated with data transfer via a memory bus 210. For increased bus loading (e.g., data transfers on a memory bus 210), the frequency spread deviation can be adjusted to correspond to a drive pad strength setting. Accordingly, a maximum pad drive strength can correspond to a maximum frequency spread deviation, a nominal pad drive strength can correspond to a nominal frequency spread deviation, and so on. Not only does the ASIC drive pad strength correspond to memory bus loading, but a relationship is established between the memory bus loading and spread spectrum clocking.

[0014] Fig. 3 illustrates a data transfer frequency chart 300 that illustrates exemplary spread spectrum clocking for data transfer bus loading. A peaked frequency pair 302 illustrates dithered clocks, or spread spectrum clocking of a frequency for a data signal. Without spread spectrum clocking of data transfer bus loading, all of the electromagnetic energy for the data signal would be concentrated at harmonics 304 (e.g., multiples of the fundamental frequency) in the frequency domain. A frequency spread deviation 306 can be controlled by adjusting a minimum clock frequency 308 and a maximum clock frequency 310 for data communication, or by adjusting a percentage clock frequency deviation from a center frequency 312.

[0015] Fig. 4 illustrates an exemplary method 400 for spread spectrum clocking for data transfer bus loading. The order in which the method is described is not intended to be construed as a limitation, and any number of the described method blocks can be combined in any order to implement the method. Furthermore, the method can be implemented in any suitable hardware, software, firmware, or combination thereof. A method for spread spectrum clocking for data transfer bus loading may also be described in the general context of computer executable instructions. Generally, computer executable instructions include routines, programs, objects, components, data structures, and the like that perform particular function(s) or implement data type(s).

[0016] At block 402, a frequency spread deviation is controlled for data transfer to a component via a data transfer bus. For example, a frequency spread deviation 306 (Fig. 3) is controlled for data transfer to a memory component 212 (Fig. 2) via memory bus 210. The frequency spread deviation can be controlled with the spread spectrum clocking control 218 (e.g., a phase-locked loop) which receives an input to adjust the frequency spread deviation from firmware component 204.

[0017] At block 404, the frequency spread deviation is adjusted to minimize electromagnetic emissions associated with data transfer via the data transfer bus. For example, a minimum clock frequency 308 (Fig. 3) and/or a maximum clock frequency 310 can be adjusted to minimize the electromagnetic transmissions associated with data transfer via a memory bus 210 (Fig. 2). Alternatively, a percentage clock frequency deviation can be adjusted from a center frequency 312 for data transfer via a memory bus 210.

[0018] At block 406, an operating conditions status is maintained that corresponds to data transfer loading on the data transfer bus. The operating conditions status corresponds to at least one of a process, a voltage, and a temperature operating condition of the data circuit 102 (Fig. 1) or of the ASIC 202 (Fig. 2). This may include maintaining a process-voltage-temperature (PVT) status in the PVT status register 216 that corresponds to data transfer loading on a memory bus 210, where the PVT status indicates the operating conditions of the ASIC 202.

[0019] At block 408, the operating conditions status is obtained from a data register. For example, a PVT status corresponding to the ASIC 202 is obtained from PVT status register 216. At block 410, a drive current strength of a variable connection drive pad is set according to the operating conditions status. For example, a drive current strength of a variable connection drive pad of the ASIC 202 is set according to the PVT status obtained from the PVT status register 216.

[0020] At block 412, an input is generated to adjust the frequency spread deviation according to the operating conditions status and/or the data transfer loading. For example, firmware 204 (Fig. 2) generates an ASIC input to the spread spectrum clocking control 218 to adjust the frequency spread deviation for data transfer. Similarly, control logic 104 (Fig. 1) generates an input to the spread spectrum clocking control 112 to adjust the frequency spread deviation for data transfer.

[0021] Fig. 5 illustrates various components of an exemplary printing device 500 in which spread spectrum clocking for data transfer bus loading can be implemented. General reference is made herein to one or more printing

devices, such as printing device 500. As used herein, "printing device" means any electronic device having data communications, data storage capabilities, and/or functions to render printed characters, text, graphics, and/or images on a print media. A printing device may be a printer, fax machine, copier, plotter, and the like. The term "printer" includes any type of printing device using a transferred imaging medium, such as ejected ink, to create an image on a print media. Examples of such a printer can include, but are not limited to, inkjet printers, electrophotographic printers, plotters, portable printing devices, as well as all-in-one, multi-function combination devices.

[0022] Printing device 500 includes one or more processors 502 (e.g., any of microprocessors, controllers, and the like) which process various instructions to control the operation of printing device 500 and to communicate with other electronic and computing devices.

[0023] Printing device 500 can be implemented with one or more memory components, examples of which include random access memory (RAM) 504, a disk drive 506, and non-volatile memory 508 (e.g., any one or more of a ROM 510, flash memory, EPROM, EEPROM, etc.). The one or more memory components store various information and/or data such as configuration information, print job information and data, graphical user interface information, fonts, templates, menu structure information, and any other types of information and data related to operational aspects of printing device 500.

[0024] Printing device 500 includes a firmware component 512 that is implemented as a permanent memory module stored on ROM 510, or with other components in printing device 500, such as a component of a processor 502. Firmware 512 is programmed and distributed with printing device 500 to

coordinate operations of the hardware within printing device 500 and contains programming constructs used to perform such operations.

[0025] An operating system 514 and one or more application programs 516 can be stored in non-volatile memory 508 and executed on processor(s) 502 to provide a runtime environment. A runtime environment facilitates extensibility of printing device 500 by allowing various interfaces to be defined that, in turn, allow application programs 516 to interact with printing device 500.

[0026] Printing device 500 further includes one or more communication interfaces 518 which can be implemented as any one or more of a serial and/or parallel interface, a wireless interface, any type of network interface, and as any other type of communication interface. A wireless interface enables printing device 500 to receive control input commands and other information from an input device, such as from an infrared (IR), 802.11, Bluetooth, or similar RF input device. A network interface provides a connection between printing device 500 and a data communication network which allows other electronic and computing devices coupled to a common data communication network to send print jobs, menu data, and other information to printing device 500 via the network. Similarly, a serial and/or parallel interface provides a data communication path directly between printing device 500 and another electronic or computing device.

[0027] Printing device 500 also includes a print unit 520 that includes mechanisms arranged to selectively apply an imaging medium such as liquid ink, toner, and the like to a print media in accordance with print data corresponding to a print job. The print media can include any form of media

used for printing such as paper, plastic, fabric, Mylar, transparencies, and the like, and different sizes and types such as 8½ x 11, A4, roll feed media, etc.

[0028] Printing device 500, when implemented as an all-in-one device for example, can also include a scan unit 522 that can be implemented as an optical scanner to produce machine-readable image data signals that are representative of a scanned image, such as a photograph or a page of printed text. The image data signals produced by scan unit 522 can be used to reproduce the scanned image on a display device or with a printing device.

[0029] Printing device 500 also includes a user interface and menu browser 524 and a display panel 526. The user interface and menu browser 524 allows a user of printing device 500 to navigate the device's menu structure. User interface 524 can be indicators or a series of buttons, switches, or other selectable controls that are manipulated by a user of the printing device. Display panel 526 is a graphical display that provides information regarding the status of printing device 500 and the current options available to a user through the menu structure.

[0030] Although shown separately, some of the components of printing device 500 can be implemented in an application specific integrated circuit (ASIC). Additionally, a system bus (not shown) typically connects the various components within printing device 500. A system bus can be implemented as one or more of any of several types of bus structures, including a memory bus or memory controller, a peripheral bus, an accelerated graphics port, or a local bus using any of a variety of bus architectures.

[0031] Fig. 6 illustrates an exemplary computing device 600 in which spread spectrum clocking for data transfer bus loading can be implemented.

Computing device 600 includes one or more processors 602 (e.g., any of microprocessors, controllers, and the like) which process various instructions to control the operation of computing device 600 and to communicate with other electronic and computing devices. Computing device 600 can be implemented with one or more memory components, examples of which include a random access memory (RAM) 604, a disk storage device 606, non-volatile memory 608 (e.g., any one or more of a read-only memory (ROM), flash memory, EPROM, EEPROM, etc.), and a floppy disk drive 610.

[0032] Disk storage device 606 can include any type of magnetic or optical storage device, such as a hard disk drive, a magnetic tape, a recordable and/or rewriteable compact disc (CD), a DVD, DVD+RW, and the like. The one or more memory components provide data storage mechanisms to store various information and/or data such as configuration information for computing device 600, graphical user interface information, and any other types of information and data related to operational aspects of computing device 600. Alternative implementations of computing device 600 can include a range of processing and memory capabilities, and may include any number of differing memory components than those illustrated in Fig. 6.

[0033] An operating system 612 and one or more application program(s) 614 can be stored in non-volatile memory 608 and executed on processor(s) 602 to provide a runtime environment. A runtime environment facilitates extensibility of computing device 600 by allowing various interfaces to be defined that, in turn, allow the application programs 614 to interact with computing device 600. The application programs 614 can include a browser to

browse the Web (e.g., "World Wide Web"), an email program to facilitate electronic mail, and any number of other different application programs.

[0034] Computing device 600 further includes one or more communication interfaces 616 and a modem 618. The communication interfaces 616 can be implemented as any one or more of a serial and/or parallel interface, as a wireless interface, any type of network interface, and as any other type of communication interface. A wireless interface enables computing device 600 to receive control input commands and other information from an input device, such as from a remote control device or from another infrared (IR), 802.11, Bluetooth, or similar RF input device.

[0035] A network interface provides a connection between computing device 600 and a data communication network which allows other electronic and computing devices coupled to a common data communication network to communicate information to computing device 600 via the network. Similarly, a serial and/or parallel interface provides a data communication path directly between computing device 600 and another electronic or computing device. Modem 618 facilitates computing device 600 communication with other electronic and computing devices via a conventional telephone line, a DSL connection, cable, and/or other type of connection.

[0036] Computing device 600 may include user input devices 620 that can include a keyboard, mouse, pointing device, and/or other mechanisms to interact with, and to input information to computing device 600. Computing device 600 also may include an integrated display device 622, such as for a potable computing device and similar mobile computing devices.

[0037] Computing device 600 also includes an audio / video processor 624 that generates display content for display on the display device 622, and generates audio content for presentation by a presentation device, such as one or more speakers (not shown). The audio / video processor 624 can include a display controller that processes the display content to display corresponding images on the display device 622. A display controller can be implemented as a graphics processor, microcontroller, integrated circuit, and/or similar video processing component to process the images. Video signals and audio signals can be communicated from computing device 600 to an external display via an RF (radio frequency) link, S-video link, composite video link, component video link, or other similar communication link.

[0038] Although shown separately, some of the components of computing device 600 may be implemented in an application specific integrated circuit (ASIC). Additionally, a system bus (not shown) typically connects the various components within computing device 600. A system bus can be implemented as one or more of any of several types of bus structures, including a memory bus or memory controller, a peripheral bus, an accelerated graphics port, or a local bus using any of a variety of bus architectures.

[0039] Although spread spectrum clocking for data transfer bus loading has been described in language specific to structural features and/or methods, it is to be understood that the subject of the appended claims is not necessarily limited to the specific features or methods described. Rather, the specific features and methods are disclosed as exemplary implementations of spread spectrum clocking for data transfer bus loading.